How to set it up:

A. DB side:

1. using PgAdmin (like we've used it throughout the curse) run the sql script in the already created twitter DB:

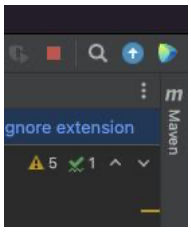
`tweets_table.sql` (this will create **tweets_t** table)

B. Java side (all of these steps can also be found in the pinned topic in Splunk (the attached PDF), as it was part of a homework):
https://kindrednukodningskurs.slack.com/archives/C02UCNRAQSJ/p1643821438385749

1. update application.properties with the DB name and credentials (the how for this is also in a pinned post on Slack):

```
-- copy-paste --
# datasource configuration (remember to start the db server and update
username and password below)
spring.datasource.url=jdbc:postgresql://localhost:5432/enterYourDbName
spring.datasource.username=postgres
spring.datasource.password=enterYourDBPassword
```

2. build the Spring Boot application by running the Maven command (can be done in IntelliJ from Maven plugin; It's on the right side of the screen – don't forget to check the PDF from pinned post):




```
mvn clean install
```

3. run the Spring Boot application by running the maven command:

```
mvn spring-boot:run
```

4. The exposed EPs can be found in the attached Postman collection:

twitter_api.postman_collection.json

Classified as General